

Amendments to the Claims:

This listing of claims will replace all prior versions, and listing, of claims in the application:

Listing of the Claims:

1. (Currently Amended) A method of loading a class file into a virtual machine, said class file being associated with a class, and said virtual machine operating in an object-oriented computing system, said method comprising:

loading said class file into a memory portion of the computing system, wherein said loading operates to copy said class file in its entirety into said memory portion prior to further processing said class file;

selecting information from said class file in said memory portion after said class file has been loaded into said memory portion, wherein said selecting operates to select at least a plurality of selected methods which are information that is to be loaded into said virtual machine; and

loading said selected information from said class file in said memory portion into said virtual machine and not loading information which has not been selected from said class file into said virtual machine;

generating an internal representation for said plurality of selected methods which are to be loaded in the virtual machine, wherein said internal representation includes a method information portion and a method reference portion, said method information portion providing information about said plurality of selected methods and said method reference portion information providing information which can be used to invoke said plurality of selected methods, and

wherein said method reference portion comprises a list, representing all of said plurality of selected methods, which is arranged as a sequence of reference cells each representing one of said plurality of selected methods, wherein each of said reference cells includes a method name field, a method signature field, and a method code field respectively providing a name, a signature, and a reference to code of a method, thereby allowing a first selected method of said selected plurality of methods to be invoked by accessing said method reference portion and identifying a first reference cell associated with said first selected method.

2-8. Cancelled

9. (Previously Presented) A method as recited in claim 1, wherein said loading of only said selected information operates to populate said method name field, method signature field, and method code field with information or references to information.

10. (Original) A method as recited in claim 1, wherein said memory is a heap memory of said computing system.

11. (Original) A method as recited in claim 1, wherein the method further comprises:
determining whether an internal representation of the class file exists in the virtual machine; and

creating an internal representation of the class file in the virtual machine when said determining determines that an internal representation of the class file does not exist in the virtual machine.

12. (Original) A method as recited in claim 1, wherein the method further comprises:
determining whether said class file exists in said memory portion; and
loading said class file in said memory portion when said determining determines that said class file does not exist in said memory portion.

13. (Original) A method as recited in claim 1, wherein the method further comprises:
removing said class file from said memory portion.

14. (Original) A method as recited in claim 13, wherein said removing is performed on a Least Recently Used basis.

15-21. (Cancelled)

22. (New) A computer readable medium including computer program code for loading a class file into a virtual machine, said class file being associated with a class, and said virtual machine operating in an object-oriented computing system, said method comprising:

computer program code for loading said class file into a memory portion of the computing system, wherein said loading operates to copy said class file in its entirety into said memory portion prior to further processing said class file;

computer program code for selecting information from said class file in said memory portion after said class file has been loaded into said memory portion, wherein said selecting operates to select at least a plurality of selected methods which are to be loaded into said virtual machine;

computer program code for loading said selected information from said class file in said memory portion into said virtual machine and not loading information which has not been selected from said class file into said virtual machine;

computer program code for generating an internal representation for said plurality of selected methods which are to be loaded in the virtual machine, wherein said internal representation includes a method information portion and a method reference portion, said method information portion providing information about said plurality of selected methods and said method reference portion information providing information which can be used to invoke said plurality of selected methods, and

wherein said method reference portion comprises a list, representing all of said plurality of selected methods, which is arranged as a sequence of reference cells each representing one of said plurality of selected methods, wherein each of said reference cells includes a method name field, a method signature field, and a method code field respectively providing a name, a signature, and a reference to code of a method, thereby allowing a first selected method of said selected plurality of methods to be invoked by accessing said method reference portion and identifying a first reference cell associated with said first selected method.

23. (New) A computer readable medium as recited in claim 22, wherein said loading of only said selected information operates to populate said method name field, method signature field, and method code field with information or references to information.

24. (New) A computer readable medium as recited in claim 22, wherein said memory is a heap memory of said computing system.

25. (New) A computer readable medium as recited in claim 22, wherein the method further comprises:

determining whether an internal representation of the class file exists in the virtual machine; and

creating an internal representation of the class file in the virtual machine when said determining determines that an internal representation of the class file does not exist in the virtual machine.

26. (New) A computer readable medium as recited in claim 22, wherein the method further comprises:

determining whether said class file exists in said memory portion; and

loading said class file in said memory portion when said determining determines that said class file does not exist in said memory portion.

27. (New) A computer readable medium as recited in claim 22, wherein the method further comprises:

removing said class file from said memory portion.

28. (New) A computer readable medium as recited in claim 22, wherein said removing is performed on a Least Recently Used basis.

29. (New) A computer system for loading a class file into a virtual machine, said class file being associated with a class of an object-oriented computer programming language, wherein said computer system comprises:

memory;

at least one processor configured to access said memory and operating to:

load a class file into a virtual machine, said class file being associated with a class, and said virtual machine operating in an object-oriented computing system, said method comprising:

load said class file into a memory portion of the computing system, wherein said loading operates to copy said class file in its entirety into said memory portion prior to further processing said class file;

select information from said class file in said memory portion after said class file has been loaded into said memory portion, wherein said selecting

operates to select at least a plurality of selected methods which are to be loaded into said virtual machine;

load said selected information from said class file in said memory portion into said virtual machine and not loading information which has not been selected from said class file into said virtual machine;

generate an internal representation for said plurality of selected methods which are to be loaded in the virtual machine, wherein said internal representation includes a method information portion and a method reference portion, said method information portion providing information about said plurality of selected methods and said method reference portion information providing information which can be used to invoke said plurality of selected methods, and

wherein said method reference portion comprises a list, representing all of said plurality of selected methods, which is arranged as a sequence of reference cells each representing one of said plurality of selected methods, wherein each of said reference cells includes a method name field, a method signature field, and a method code field respectively providing a name, a signature, and a reference to code of a method, thereby allowing a first selected method of said selected plurality of methods to be invoked by accessing said method reference portion and identifying a first reference cell associated with said first selected method.

30. (New) A computer system as recited in claim 29, wherein said loading of only said selected information operates to populate said method name field, method signature field, and method code field with information or references to information.

31. (New) A computer system as recited in claim 29, wherein said memory is a heap memory of said computing system.

32. (New) A computer system as recited in claim 29, wherein the method further comprises:

determining whether an internal representation of the class file exists in the virtual machine; and

creating an internal representation of the class file in the virtual machine when said determining determines that an internal representation of the class file does not exist in the virtual machine.

33. (New) A computer system as recited in claim 29, wherein the method further comprises:

 determining whether said class file exists in said memory portion; and
 loading said class file in said memory portion when said determining determines that said class file does not exist in said memory portion.

34. (New) A computer system as recited in claim 29 wherein the method further comprises:

 removing said class file from said memory portion.

35. (New) A method as recited in claim 34, wherein said removing is performed on a Least Recently Used basis.